

## **REMARKS/AGRUMENTS**

### **1. Summary of the Office Action**

Claims 1-29 stand rejected under 35 U.S.C. §102(b) as allegedly being anticipated by Bach et al. (US patent no. 6,141,660).

Bach is directed at a method for generating class specifications for an object-oriented application program that accesses a hierarchical database. (Bach, 5: 35-37.)

### **2. Response to § 102 Rejections**

Applicant respectfully traverses this rejection for the reasons set out below, and ask the Examiner for reconsideration.

To anticipate a claim, the reference must teach every element of the claim. “A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

- i. BACH FAILS TO TEACH THE FEATURE OF “CREATING A CLI DESCRIPTION FILE FOR EACH COMMAND IN THE SET OF COMMANDS TO ABSTRACT THE CLI FROM THE SYSTEM” AS REQUIRED BY CLAIM**

**1**

Claim 1 reads as follows:

1. A method, including:
  - defining a set of commands to be used with a command line interface (CLI), each command in the set of commands specifying an action to be performed in a system;
  - defining a set of system interfaces including objects and methods, wherein each action specified in the command is associated with an object and an method, the object and the method performing semantics required by the command; and
  - creating a CLI description file for each command in the set of commands to abstract the CLI from the system, the CLI description file mapping the command to be used with the CLI with the action and the associated objects and the associated methods,****wherein the command is to be entered into the system in text mode.**

Bach discloses the following:

*The class specifications are generated using a command line interface of a class definition tool. The class definition tool parses database files and generates class definitions for objects that encapsulate or wrapper data retrieved from the database. The class definition tool also automatically generates input forms and output pages (for example, HTML or XML forms and pages) that are displayed on web browsers that interact with the application program and objects framework.*

Bach, 5: 37-45.

Bach fails to disclose or even suggest a feature of “**creating a CLI description file for each command in the set of commands to abstract the CLI from the system,** the CLI description file mapping the command to be used with the CLI with the action and the associated objects and the associated methods, wherein the command is to be entered into the system in text mode”, as required by claim 1.

In Bach, there is no mention of a CLI description file and no indication of a mechanism to abstract the command line interface (CLI) from the system. In contrast, the CLI in Bach may be relying on a set of closely coupled functions between the commands defined and the class definition tool which is distinct from **“creating a CLI description file for each command in the set of commands to abstract the CLI from the system”**, as required by claim 1.

ii. **THE LANGUAGE IN BACH RELIED UPON BY THE OFFICE ACTION IS UNRELATED TO THE FEATURE OF “CREATING A CLI DESCRIPTION FILE FOR EACH COMMAND IN THE SET OF COMMANDS TO ABSTRACT THE CLI FROM THE SYSTEM” OF CLAIM 1**

The Detailed Action relies on “Bach, 8: 1-23” and “Bach, 12: 1-19” to show the abovementioned feature of claim 1 (Official Paper 9, p. 2). However, the passage in “Bach, 8: 1-23” relates to objects framework, which resides on the server 102 (Bach, Fig. 1) and is *unrelated to the class definer tool (CDT) and its CLI, both of which reside on the client 100* (Bach, Fig. 1, Fig. 4). The passage in “Bach, 12: 1-19” relates to a Graphical User Interface (GUI), as opposed to Command Line Interface (CLI), and therefore is not relevant to the feature of **“creating a CLI description file for each command in the set of commands to abstract the CLI from the system, the CLI description file mapping the command to be used with the CLI with the action and the associated objects and the associated methods, wherein the command is to be entered into the system in text mode”** of claim 1.

iii. **BACH DOES NOT TEACH EVERY ELEMENT OF CLAIMS 12 AND 23**

Claims 12 and 23 include the feature of **“creating a CLI description file for each command in the set of commands to abstract the CLI from the system, the CLI description file mapping the command to be used with the CLI with the action and the associated objects and the associated methods, wherein the command is to be entered into**

the system in text mode". Thus, claims 12 and 23 are patentable for at least the reasons articulated with respect to claim 1.



**iv. BACH DOES NOT TEACH EVERY ELEMENT OF CLAIMS 2-11, 13-22,  
AND 24-29**

As a dependent claim is deemed to include the limitations of a claim from which it depends, the arguments presented above also address the rejections against the dependent claims. Accordingly, the rejections against the dependent claims have been addressed, and withdrawal of these rejections is respectfully requested.

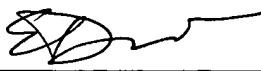
**3. Conclusion**

Having tendered the above remarks and amended the claims as indicated herein, Applicant respectfully submits that all rejections have been addressed and that the claims are now in a condition for allowance, which is earnestly solicited. Authorization is hereby given to charge our Deposit Account No. 02-2666 for any charges that may be due. Furthermore, if an extension is required, then Applicants hereby request such an extension.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: 5/21-, 2004

  
\_\_\_\_\_  
Elena B. Dreszer  
Reg. No. 55,128

12400 Wilshire Blvd.  
Seventh Floor  
Los Angeles, CA 90025-1026  
(408) 947-8200